

SCPI 通信协议

型号IT6322

三路可编程直流电源

第一章：通讯模块介绍	4
1.1 IT-E131 RS232通讯电缆	4
1.2 IT-E132 USB通讯电缆	4
1.3 IT-E135 GPIB通讯电缆	5
第二章 SCPI 命令表	6
2.1 IEEE488.2 共同命令	6
2.2 SCPI 必备命令	6
2.3 非SCPI标准命令	8
第三章 SCPI 状态寄存器	9
第四章 SCPI 命令描述	11
4.1 IEEE488.2 共同命令	11
4.2 SCPI 必备命令	13

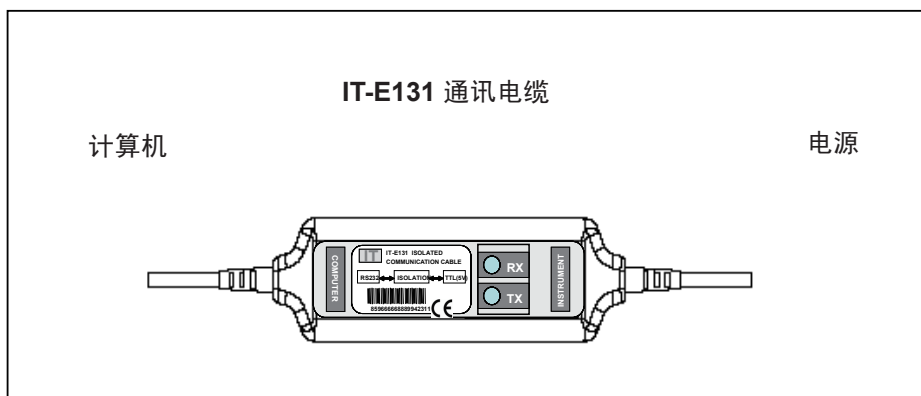
通告

本手册的内容如有更改，恕不另行通知。

电源供应器能够通过后面板上的DB9口经通讯电缆连接到计算机的相应接口上，本协议适用与以下通讯电缆。

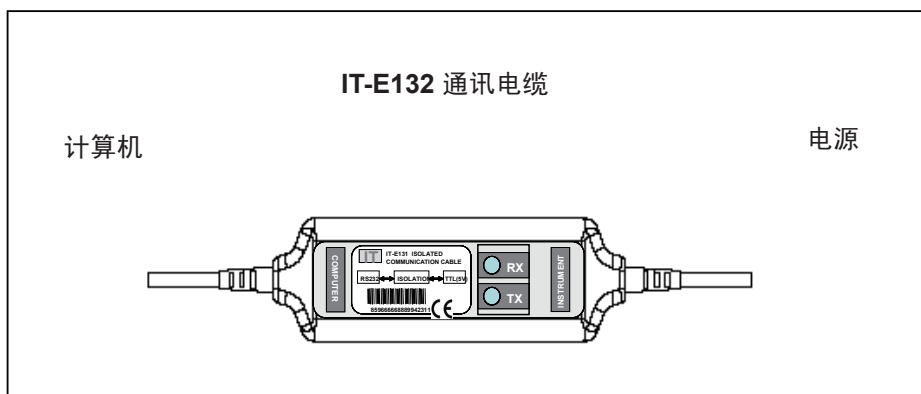
1.1 IT-E131 RS232通讯电缆

电源供应器后面板的DB9接口输出为TTL电平，您需要通过附件IT-E131通讯电缆转换后才可以连接到计算机的串口上。IT-E131的连接脚说明如下



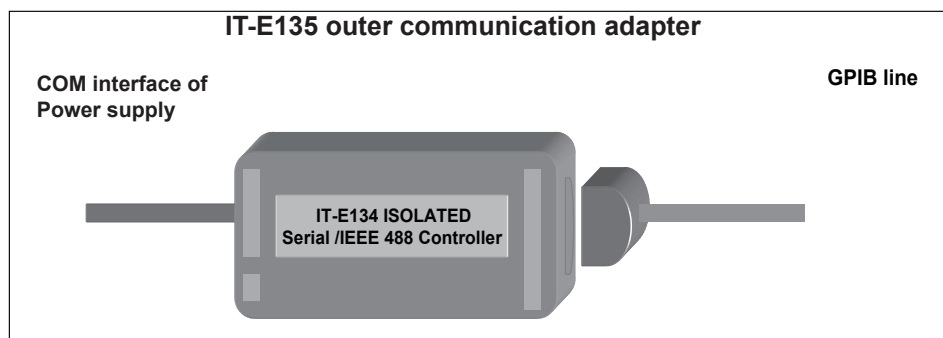
1.2 IT-E132 USB通讯电缆

电源供应器后面板的DB9接口输出为TTL电平，您需要通过附件IT-E132通讯电缆转换后才可以连接到计算机的USB接口上。产品大小和外观除接口处为USB接口，其他均和上面IT-E131相同。



1.3 IT-E135 GPIB通讯电缆

此为一个外接卡形式的GPIB接口，内部使用的是NI公司通用的芯片，您需要先将电源的DB9接口与IT-E135通讯盒相接，再通过一根GPIB/IEEE488线缆将IT-E135通讯盒和计算机的GPIB接口相连。



注意不能把电源的DB9通讯接口直接用标准RS-232/USB/GPIB电缆连接到PC的串口或RS232电平上，必须使用我公司的通讯电缆来连接。

2.1 IEEE488.2 共同命令

"*CLS"
"*ESE"
"*ESE?"
"*ESR?",
"*IDN?",
"*OPC",
"*OPC?",
"*PSC",
"*PSC? ",
"*RST",
"*SRE",
"*SRE?",
"*STB?",
"*TRG",
"*SAV ",
"*RCL",

2.2 SCPI 必备命令

SYSTem
:ERRor?
:VERSion?
:BEEPer[:IMMediate]
:ADDRes?

STATus
:QUESTionable
:ENABle <enable value>
:ENABle?
[:EVENT]?
:CONDition?
:OPERation
:ENABle <enable value>
:ENABle?
[:EVENT]?
:CONDition?
:INSTrument
[:EVENT]?
:ENABle <value>
:ENABle?
CONDition?

[:EVENT]?
:ENABle <value>
:ENABle?
CONDition?

INSTrument

[:SElect] {FIRst|SECOnd|THIrd}
[:SElect]?
NSElect {1|2|3}
NSElect?

OUTPut

[:STATe] {0|1}
[:STATe]?

[SOURce:]

CURRent[:LEVel][:IMMediate][:AMPLitude] {<current>|MIN|MAX}
CURRent[:LEVel][:IMMediate][:AMPLitude]? {MIN|MAX}
VOLTage[:LEVel][:IMMediate][:AMPLitude] {<voltage>|MIN|MAX}
VOLTage[:LEVel][:IMMediate][:AMPLitude]? {MIN|MAX}
VOLTage:PROTection[:LEVel][:IMMediate][:AMPLitude]
VOLTage:PROTection[:LEVel][:IMMediate][:AMPLitude]?

2.3 非SCPI标准命令

CALibration

```
:SECure[:STATe] {ON|OFF,<quoted code>}  
:SECure[:STATe]?  
:VOLTage  
    LEVel <level>  
    [:DATA] <voltage value>  
:CURRent  
    LEVel <level>  
    [:DATA] <current value>  
:CODE  
:SAVe  
:INITial
```

OUTPut

```
:TIMer  
:DATA <time>  
:DATA?
```

SYSTem

```
:LOCal  
:REMote  
:RWLock
```

DISPlay:

```
[:WINDow][:STATe] {OFF|ON}  
[:WINDow][:STATe]?
```

MEASure[:SCALer]

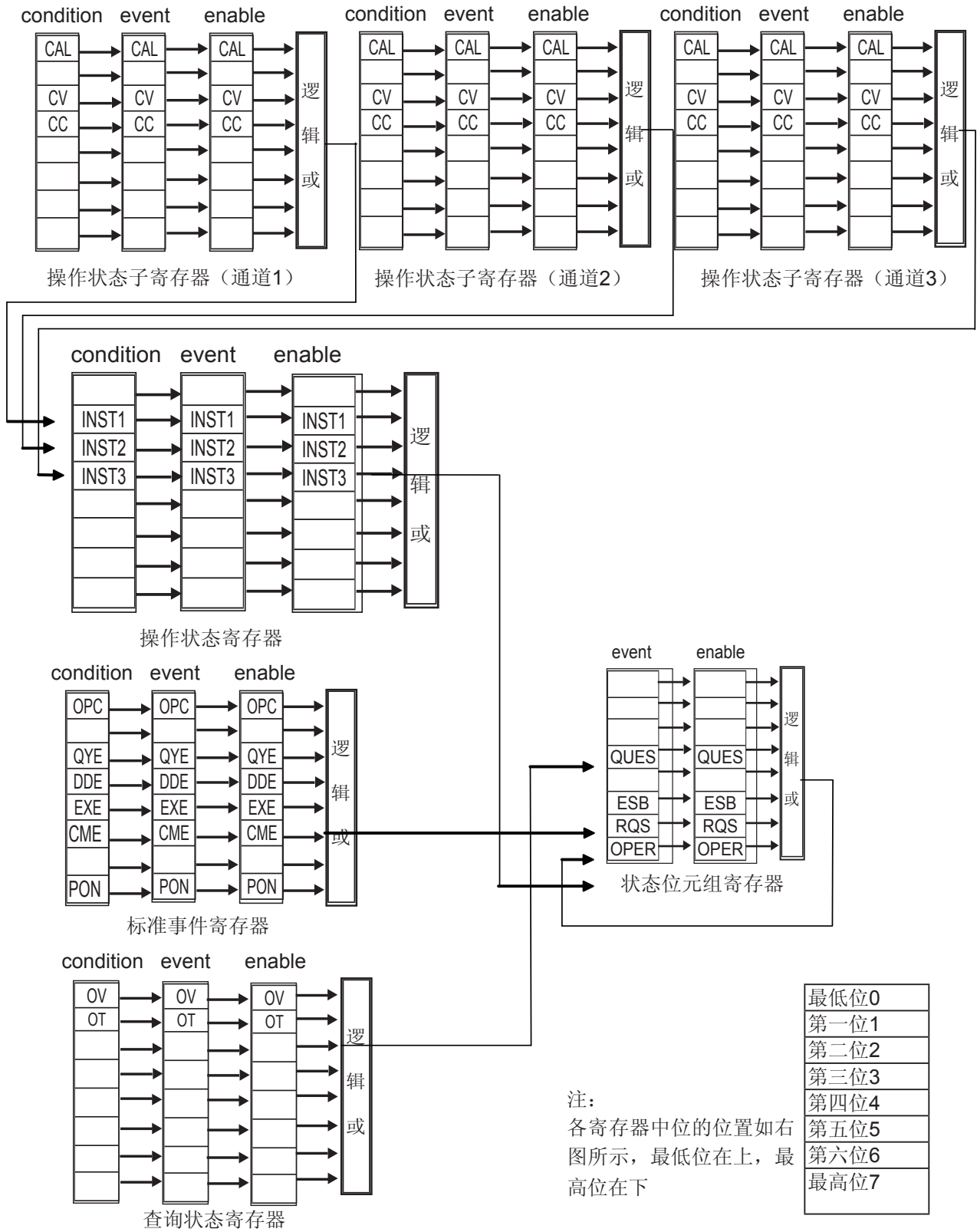
```
:CURRent[:DC]?  
:VOLTage[:DC]?  
:POWER[:DC]?
```


你可以通过读取操作状态寄存器的值来确定电源的当前状态。电源通过七个状态寄存器组记录了不同的仪器状态，这七个状态寄存器组分为状态位组寄存器，标准事件寄存器，查询状态寄存器和操作状态寄存器（包含三个子寄存器）。状态位组寄存器记录了其它状态寄存器的信息。

下表给出了各个状态寄存器的定义。

	位	代号	含义
操作状态子寄存器	0	CAL	电源正在计算新的标定系数
	2	CV	电源正处在定电压输出状态
	3	CC	电源正处在定电流输出状态
	4	RI	未使用
查询状态寄存器	0	OV	未使用
	1	OT	电源处于过温度状态
标准状态寄存器	0	OPC	操作完成，电源并行操作被完成
	2	QYE	查询错误，输出队列数据丢失
	3	DDE	仪器相关错误
	4	EXE	执行错误，命令参数溢出或操作条件不一致
	5	CME	命令错误，在接受的命令信息中有语法或语义错误。
状态字节寄存器	7	PON	开机位：开机后该位为1
	3	QUES	如果一个使能的查询状态寄存器的状态发生变化，则QUES位置1
	4		未使用
	5	ESB	若一个使能的标准事件状态寄存器的状态发生变化，则ESB位置1
	6	MSS	自身
操作状态寄存器	7	OPER	若一个使能的操作状态寄存器的状态发生变化，则OPER位置1
	1	INST1	如果一个操作状态子寄存器的状态发生变化，则INST1位置1
	2	INST2	同上
	3	INST3	同上

下图定义了电源状态寄存器的结构



4.1 IEEE488.2 共同命令

*CLS

这条命令清除下面的寄存器：

标准事件寄存器, 查询事件寄存器, 操作状态寄存器, 操作状态子寄存器, 状态字节寄存器的错误信息

命令语法: *CLS

例: *CLS

*ESE

这条命令编辑了标准事件使能寄存器的值。

编程参数决定了标准事件寄存器中哪些位为1时将会引起状态位组寄存器中ESB位置1。

命令语法: *ESE <NRf>

参数: 0~255

上电值: 参考*PSC命令

例: *ESE 110

*ESR?

这条命令可以用来读取标准事件寄存器的值。在该命令被执行后, 标准事件寄存器的值被清零。标准事件寄存器的位定义与标准事件使能寄存器的位定义相同。

命令语法: *ESR?

返回参数: <NR1>

例: *ESR?

*IDN?

这条命令可以读电源的相关信息, 它返回的参数包含了四个段。

命令语法: *IDN?

返回参数: " 制造商 ", " 产品标号 ", " 产品序列号 ", " 软件版本号 "。

例: ITECH,IT6322,005004156269001029,V1.69

*OPC

当在这条命令这前的所有命令被执行完成后, 标准事件寄存器的OPC位被置1。

命令语法: *OPC

上电值: 参考*PSC命令。

例: *OPC

*PSC

该命令用来控制当电源上电时是否会产生一个服务请求。

1|ON: 当电源上电时, 所有使能寄存器的值被清零。

0|OFF: 当电源上电时。所有使能寄存器的值为上次保存的值。

命令语法: *PSC <参数>

参数: 0|1|ON|OFF

上电值: 上次保存的值。

例: *PSC ON

*RST

这条命令复位电源所有的参数到出厂状态。

命令语法: *RST

例: *RST

*SRE

这条命令编辑了状态位组使能寄存器的值。

编程参数决定了状态位组寄存器中哪些位为1时将会引起状态位组寄存器中RQS位置1。

状态位组使能寄存器的位定义与状态位组寄存器的位定义相同。

命令语法: *SRE <参数>

参数: 0~255

上电值: 参考*PSC命令。

例: *SRE 110

*STB?

这条命令用来读取状态字节寄存器的值。

该命令被执行后, 状态字节寄存器的值被清零。

命令语法: *STB?

返回参数: <NR1>

例: *STB?

*SAV

这条命令用来保存电源的当前用户设定值(当前电流、电压、最大电压、步进位)到指定单元。

命令语法: *SAV <参数>

参数: 0~49

例: *SAV 10

*RCL

这条命令将从指定单元中恢复*SAV命令保存的设定值。

命令语法: *RCL <参数>

参数: 0~49

例: *RCL 10

4.2 SCPI 必备命令

SYSTem:ERRor

这条命令用来读取电源的出错信息。

命令语法：SYSTem:ERRor?

返回参数：参见下表

例：SYST:ERR?

代码		含义
0	'No Error'	没有错误
1	'Too Many Num Suf'	数据缓冲中有太多的数据无法处理
10	'No Command'	无效命令
14	'Num Suf Invalid'	数字的下标识非法值
16	'Invalid Number'	非法数据
17	'Invalid Dims'	无效的数据维数
20	'Param Overflow'	参数溢出
30	'Error Para Units'	参数单位错误
40	'Error Para Type'	参数类型错误
50	'Error Para Count'	参数个数错误
60	'Unmatched Quote'	参数中引用标记不匹配
65	'Unmatch. Bracket'	参数缺少括号
70	'Invalid Command'	非法命令
80	'No Entry'	找不到命令入口
90	'Too Many Dims'	数据维数过多
100	'Too Many Command'	命令过多
101	'Command Exec Err'	命令执行错误
110	'Error Rxd Parity'	奇偶校验错误
120	'Error EEPROM'	EEPROM检测出错
121	'Error Config Data'	配置数据错误
122	'Error Cal. Data'	校准数据错误
123	'Error Factory Data'	厂方校准数据出错

SYSTem:VERSion

这条命令用来查询软件的版本号。

命令语法: SYSTem:VERSion?

返回参数: 软件的版本号。

例: SYST:VERS?

SYSTem:BEEPer

这条命令用来测试蜂鸣器, 执行后电源应鸣叫一声。

命令语法: SYSTem:BEEPer[:IMMEDIATE]

例: SYST:BEEP

SYSTem:LOCAl

这条命令用来设置电源为本地操作状态。

命令语法: SYSTem:LOCAl

例: SYST:LOC

SYSTem:REMote

这条命令用来设置电源为远程操作状态。

命令语法: SYSTem:REMote

例: SYST:REM

SYST:RWLock

这条命令也是用来设置电源为远程操作状态的。

但与上一条不同的是, 本命令无法在电源上按(LOCATE)键切换到本地操作状态, 需用命令才能恢复。

命令语法: SYSTem:RWLock

例: SYST:RWL

SYSTem:ADDRess

这条命令用来查询电源的本机地址。

命令语法: SYSTem:ADDRess?

例: SYST:ADDR?

STATus:QUEStionable:ENABLE

这条命令编辑了查询事件使能寄存器的值。

编程参数决定了查询事件寄存器中哪些位为1时将会引起状态位组寄存器中QUES位置1

命令语法: STATus:QUEStionable:ENABLE <参数>

参数: 0~255

上电值: 参考PSC命令。

例: STAT: QUES: ENAB 110

STATus:QUEStionable:ENABle?

这条命令用来读取查询事件使能寄存器的值。

该命令被执行后，查询事件使能寄存器的值被清零。

命令语法：STATus:QUEStionable:ENABle?

返回参数：<NR1>

例：STAT:QUES:ENAB?

STATus:QUEStionable?

这条命令可以用来读取查询事件寄存器的值。

在该命令被执行后，查询事件寄存器的值被清零。

命令语法：STATus:QUEStionable[:EVENTt]?

返回参数：<NR1>。

例：STAT:QUES?

STATus:QUEStionable:CONDition?

这条命令可以用来读取查询条件寄存器的值。

当查询条件寄存器中某位的值变化时，则查询事件寄存器中对应的位被置1

命令语法：STATus:QUEStionable:CONDition?

返回参数：<NR1>

例：STAT:QUES:COND?

STATus:OPERation ENABle

这条命令编辑了操作事件使能寄存器的值。

编程参数决定了操作事件寄存器中哪些位为1时将会引起状态位组寄存器中OPER位置1。

命令语法：STATus:OPERation:ENABle <参数>

参数：0~255

上电值：参考PSC命令。

例：STAT:OPER:ENAB 110

STATus:OPERation:ENABle?

这条命令可以用来读取操作使能寄存器的值。

在该命令被执行后，操作使能寄存器的值被清零。

查询语法：STATus:OPERation:ENABle?

返回参数：<操作使能寄存器的值>

例：STAT:OPER:ENAB?

STATus:OPERation?

这条命令可以用来读取操作事件寄存器的值。

在该命令被执行后，操作事件寄存器的值被清零。

查询语法：STATus:OPERation[:EVENTt]?

返回参数：<操作事件寄存器的值>

例：STAT:OPER?

STATus:OPERation:INSTrument?

这条命令用来读取操作事件子寄存器的值。

在该命令被执行后，操作事件子寄存器的值被清零。（注意：仅对当前通道的寄存器有效）

查询语法：STATus:OPERation:INSTrument[:EVENTj]?

返回参数：〈操作事件子寄存器的值〉

例：STAT:OPER:INST?

STATus:OPERation:INSTrument:ENABle

这条命令用来设置操作事件使能子寄存器的值，编程参数决定了操作事件使能子寄存器中哪些位为1时将会引起状态字节寄存器中OPER位位置1。（注意：仅对当前通道的寄存器有效）

命令语法：STATus:OPERation:INSTrument:ENABle <value>

参数：0~255

上电值：参考PSC命令。

例：STAT:OPER:INST:ENAB 110

STATus:OPERation:INSTrument:ENABle?

这条命令用来读取操作使能子寄存器的值。

在该命令被执行后，操作使能子寄存器的值被清零。（注意：仅对当前通道的寄存器有效）

查询语法：STATus:OPERation:INSTrument:ENABle?

返回参数：〈操作事件使能子寄存器的值〉

例：STAT:OPER:INST:ENAB?

STATus:OPERation:INSTrument:CONDition?

这条命令用来读取操作条件子寄存器的值，在该命令被执行后，操作条件子寄存器的值被清零。（注意：仅对当前通道的寄存器有效）

查询语法：STATus:OPERation:INSTrument:CONDition?

返回参数：〈操作条件寄存器的值〉

例：STAT:OPER:INST:COND?

INSTrument[SELect]

这条命令用来选择当前的通道。

命令语法：INSTrument[:SELect] <参数>

参数：FIRst|SECOnd|THIrd，表示三个通道。

上电值：FIRst

例：INST SECO

INSTrument[:SELect]?

这条命令用来读取当前选择的通道。

查询语法：INSTrument[:SELect]?

返回参数：FIRst(第一通道)|SECOnd(第二通道)|THIrd(第三通道)

例：INST?

INSTrument:NSElect

这条命令与INSTrument[SElect]命令相似，不同仅在用数字表示通道。

命令语法：INSTrument:NSElect <参数>

参数：1~3

上电值：1

例：INST: NSEL?

INSTrument:NSElect?

这条命令用来读取当前选择的通道。

它与INSTrument[:NSElect]不同仅在用数字表示通道。

查询语法：INSTrument:NSElect?

返回参数：1~3

例：INST: NSEL?

OUTPut[:STATe]

这条命令用来设置电源的当前通道为开启和关闭状态。

命令语法：OUTPut[:STATe] <参数>

参数：1|0/ON/OFF

例：OUTP 1

OUTPut[:STATe]?

这条命令用来查询电源的当前通道为开启和关闭状态。

查询语法：OUTPut[:STATe]?

返回参数：0|1

例：OUTP?

OUTPut:TIMer:DATA

这条命令用来设置电源的当前通道的输出定时器的时间。

命令语法：OUTPut:TIMer:DATA <参数>

参数：0~999999秒。(0秒为关闭定时器)。

上电值：与菜单设置有关。

例：OUTPut:TIMer:DATA 100

OUTPut:TIMer:DATA?

这条命令用来查询电源的当前通道的输出定时器的时间。

查询语法：OUTPut:TIMer:DATA?

返回参数：0~999999

单位：秒。

例：OUTP:TIM:DATA?

[SOURce:]CURRent[:LEVel][:IMMediate][:AMPLitude]

这条命令用来设置电源的当前通道的输出电流值。

命令语法: [SOURce:]CURRent[:LEVel][:IMMediate][:AMPLitude] <参数>

参数: MIN|MAX|MIN TO MAX

单位: A mA uA

上电值: 参考菜单设置。

复位值: MAX

例: CURR 2A

[SOURce:]CURRent[:LEVel][:IMMediate][:AMPLitude]?

这条命令用来查询电源的当前通道的设置电流。

查询语法: [SOURce:]CURRent[:LEVel][:IMMediate][:AMPLitude]? <参数>

参数: MIN | MAX |无

返回参数: MIN TO MAX

单位: A

例: CURR?

[SOURce:]VOLTage[:LEVel][:IMMediate][:AMPLitude]

这条命令用来设置电源的当前通道的输出电压值。

命令语法: [SOURce:]VOLTage[:LEVel][:IMMediate][:AMPLitude]

参数: MIN|MAX|MIN TO MAX

单位: V mV uV kV

上电值: 参考菜单设置。

复位值: MIN

例: VOLT 10V

[SOURce:]VOLTage[:LEVel][:IMMediate][:AMPLitude]?

这条命令用来查询电源的当前通道的设置电压值。

命令语法: [SOURce:]VOLTage[:LEVel][:IMMediate][:AMPLitude]? <参数>

参数: MIN|MAX|无

返回参数: MIN TO MAX

单位: V

例: VOLT?

[SOURce:]VOLTage:PROTection[:LEVel][:IMMediate][:AMPLitude]

这条命令用来设置电源的当前通道的最大输出电压值。

命令语法: [SOURce:]VOLTage:PROTection[:LEVel][:IMMediate][:AMPLitude] <参数>

参数: 不超过电源所能输出的最大电压。

单位: V mV kV uV

上电值: 参考菜单设置。

复位值: 电源所能输出的最大电压。

例: VOLT:PROT 20V

[SOURce:]VOLTage:PROTection[:LEVel][:IMMEDIATE][:AMPLitude]?

这条命令用来查询电源的当前通道的最大设置电压。

命令语法: **[SOURce:]VOLTage:PROTection[:LEVel][:IMMEDIATE][:AMPLitude]?**

返回参数: 电源的当前通道的最大输出电压值

单位: V

例: **VOLT:PROT?**

APPLy:VOLTage[:LEVel][:IMMEDIATE][:AMPLitude]

这条命令用来同时设置三通道设定电压。

参数类型: 三个通道的设定电压值。依次为一、二、三通道。

返回值: 无。

例: **APP:VOLT 1,2,3** (分别设置三路电压为1V、2V、3V)

APPLy:VOLTage[:LEVel][:IMMEDIATE][:AMPLitude]?

这条命令用来同时读取三通道设定电压。

参数类型: 无

返回值: 三路设定电压值。

例: **APP:VOLT?**

APPLy:CURRent[:LEVel][:IMMEDIATE][:AMPLitude]

这条命令用来同时设置三通道设定电流。

参数类型: 三个通道的设定电流值。依次为一、二、三通道。

返回值: 无。

例: **APP:CURR 1,2,3** (分别设置三路电流为1A、2A、3A)

APPLy:CURRent[:LEVel][:IMMEDIATE][:AMPLitude]?

这条命令用来同时读取三通道设定电流。

参数类型: 无

返回值: 三路设定电流值。

例: **APP:CURR?**

APPLy:PROToltage[:LEVel][:IMMEDIATE][:AMPLitude]

这条命令用来同时设置三通道设定最大电压。

参数类型: 三个通道的设定最大电压值。依次为一、二、三通道。

返回值: 无。

例: **APP:PROT 1,2,3** (分别设置三路最大电压为1V、2V、3V)

APPLy:PROToltage[:LEVel][:IMMEDIATE][:AMPLitude]?

这条命令用来同时读取三通道最大电压。

参数类型: 无

返回值: 三路设定电压值。

例: **APP:PROT?**

APPLy:OUT[:STATe]

这条命令用来同时控制三通道ON/OFF状态。

参数类型：0|1|ON|OFF

返回值：无

例：APP:OUT ON, OFF, ON

APP:OUT 1,0,1

APPLy:OUT[:STATe]?

这条命令用来同时读取三通道ON/OFF状态。

参数类型：无

返回值：0|1|ON|OFF

例：APP:OUT?

MEASure[:SCALer]:CURRent[:DC]?

读取实际输出电流。

命令语法：MEASure[:SCALer]:CURRent[:DC]?

返回参数：实际输出电流。

单位：A

例：MEAS: CURR?

MEAS[:SCALer][:VOLTage][:DC]?

读取实际输出电压。

命令语法：MEAS[:SCALer][:VOLTage][:DC]?

返回参数：实际输出电压。

单位：V

例：MEAS?

MEASure[:SCALer]:POWER[:DC]?

读取实际输出功率。

命令语法：MEASure[:SCALer]:POWER[:DC]?

返回参数：实际输出功率。

单位：W

例：MEAS:POW?

MEASure[:SCALer]:VOLTage:ALL[:DC]?

功能：同时查询三信道的实际输出电压值。

参数类型：无

返回值：三信道实际电压值。

示例：MEAS:VOLT:ALL?

MEASure[:SCALar]:CURRent:ALL[:DC]?

功能：同时查询三通道的实际输出电流值。

参数类型：无

返回值：三通道实际电流值。

示例：MEAS:CURR:ALL?

MEASure[:SCALar]:POWER:ALL[:DC]?

功能：同时查询三通道的输出功率

参数类型：无

返回值：三通道实际输出功率。

示例：MEAS:POW:ALL?

DISPlay[:WINDow][:STATe]

开启显示屏/关闭显示屏。

命令语法：DISPlay[:WINDow][:STATe] <参数>

参数：0(关闭) | 1(开启)

上电值：1(开启)

复位值：1(开启)

例：DISP 1

DISPlay[:WINDow][:STATe]?

开启显示屏/关闭显示屏状态查询。

返回参数：1(开启) | 0(关闭)

例：DISP?

CALibration:SECure[:STATe]

关闭/开启校准保护，当校准保护被禁止后才能进行校准且在校准保护被禁止后命令表中除标准命令外其它的命令都不能用，且仅能校准当前通道。

命令语法：CALibration:SECure[:STATe] <参数1>, <参数2>

参数1：0(关闭) | 1(开启)

参数2：校验密码

单位：无。

上电值：1 | (开启)

复位值：1 | (开启)

例：CAL: SEC 0,"6322"

CALibration:SECure[:STATe]?

读取校准保护位的状态。

查询语法：CALibration:SECure[:STATe]?

返回参数：0(关闭) | 1(开启)

例：CAL:SEC?

CALibration:VOLTage:LEVel

设置电压校准的校准点。

命令语法: **CALibration:VOLTage:LEVel** <参数>

参数: P1<第1点>|P2<第2点>

例: **CAL:VOLT P1**

CALibration:VOLT[:DATA]

设置电压校准的校准点电压。

命令语法: **CALibration:VOLT[:DATA]** <参数>

参数: 当前的实际输出电压。

单位: V mV uV kV

例: **CAL:VOLT 1V**

CALibration:CURREnt:LEVel

设置电流校准的校准点。

命令语法: **CALibration:CURREnt:LEVel** <参数>

参数: P1<第1点>|P2<第2点>

例: **CAL:CURR:LEV P1**

CALibration:CURREnt[:DATA]

设置电流校准的校准点电流。

命令语法: **CALibration:CURREnt[:DATA]** <参数>

参数: P1<第1点>|P2<第2点>

例: **CAL:CURR 0.3A**

CALibration:SECure:CODE

设置新的校准密码

命令语法: **CALibration:SECure:CODE** <参数>

参数: 长度为4的校准密码字符串。

例: **CAL:SEC:CODE "1234"**

CALibration:STRing

设置校准时的校准信息。

命令语法: **CALibration:STRing** <参数>

参数: 最大长度为24个字母的字符串, 也就是用户校准时记录的相关信息。

如校准时的时间、次数等。

例: **CAL:STR "2005-1-9 20:12"**

CALibration:STRing?

查看当时的校准信息。

查询语法: **CALibration:STRing?**

返回参数: 保存在电源中的校准信息

例: **CAL:STR?**

CALibration:SAVe

保存校准系数到EEPROM

命令语法: CALibration:SAVe

例: CAL:SAV

说明: 校准后的校准数据只有保存后才会下次生效。

CALibration:INITal

初始化校准系数, 恢复到出厂值。

命令语法: CALibration:INITal

例: CAL:INIT

说明: 当校准失败后可用此命令恢复。



艾德克斯电子（南京）有限公司

中国

电话： 025-52415098

传真： 025-52415268

中国南京市宁南大道310号

欧洲

电话： 477-590101

传真： 477-572323

32, rue Edouard Martel

42100 - St Etienne France

美国

电话： 714-9219095

传真： 714-9216422

22820 Savi Ranch Parkway

Yorba Linda, CA 92887 U.S.A.

韩国

电话： 285-20680

传真： 285-20684

#153-783,Rm601,ByuckSan,

GaSan-Dong, Seoul, Korea